

11. Yash

Program Name: Yash.java

Input File: yash.dat

Yash has just learned in computer science class about **Order of Magnitude**, or **Big O**, and wants to work out the numbers. For example, he has learned that any algorithm with an efficiency of $O(1)$ generally takes 1 step to complete, regardless of the size of the data, with varying larger values for the other levels of efficiency, such as $O(\log N)$, $O(N)$, $O(N\log N)$, and $O(N^2)$. He learned in class that for a data set of 10 items, these five values are 1, 4, 10, 40, and 100. He was bit confused by $O(\log N)$, until his teacher said, *"Think about the exponent for the power of 2 that equals or just exceeds 10."* He thought, *"OK, 2^1 is 2, 2^2 is 4, 2^3 is 8, and 2^4 is 16. That's why 4 is the log base 2, or $O(\log N)$, answer for the value 10! I get it! It's just the integer exponent of the number using base 2, that creates a value equal to or just past the number."*

He then tried to work out higher values of N , but started to get confused again, and needs your help.

Input: Several integers N , each on one line, representing the number of elements of data to be processed by an algorithm.

Output: The five values associated with five levels of Big O algorithm efficiency, as described above, for each value N , with values exceeding 999 shown using comma separation, and a single space between each value.

Sample Input:

```
10
50
100
```

Sample Output:

```
1 4 10 40 100
1 6 50 300 2,500
1 7 100 700 10,000
```